CSCI-1411 Fundamentals Of Computing Lab

Anh Nguyen

Fall 2015

Lab 2: Introduction to the C++ Programming Language

- Overview:
  - Lab 2 Components
    - Lab Sections (2.1, 2.2, 2.3, 2.4)
  - Lab 2 Concepts
    - Printing
    - Constants
    - Arithmetic (C++ data-types, operators)
      - `int` – Integer type
      - `float, double` – Floating point (decimal representations)
    - Characters and Strings
      - `char` – Represents a single character
      - `string` – Represents a sequence of characters (textual information)

Lab 2: Introduction to the C++ Programming Language

- C++ Constants
  - Defines a constant that will not change during execution
    - Ex: e, pi, etc
  - Removes ‘magic numbers’ from code:

```
int value = 4 * 5; // What is value? What does 4 represent? 5?

const int RECTANLGE_WIDTH = 4;  // For some reason the rectangle
const int RECTANGLE_HEIGHT = 5; // dimensions never change.
int area = RECTANGLE_WIDTH * RECTANLGE_HEIGHT; // Rectangle Area
```

  - Makes code explicit and easier to read
    - Again, it’s for humans, the computer doesn’t care

Lab 2: Introduction to the C++ Programming Language

- C++ Arithmetic
  - Similar to mathematical notation
    - You would generally write: x = 4, and alternatively: 4 = x
    - Only one of these is valid in C++
      - `int value = 4; // Valid`
      - `4 = int value; // Invalid (Maybe in English the equivalent would be something like “Four is an integer value”, but to C++ this is invalid syntax`
    - Ensure to properly space your equations:
      - `int value = 44*4-3+3(3+4/4)-2*(3);`
      - `int value = 44 * (4 – 3) + 3 * ((3 + 4) / 4) – 2 * 3;`
  - Compiler is always the ultimate test (for syntax)

Lab 2: Introduction to the C++ Programming Language

- C++ Characters
  - Characters are a ‘built-in’ primitive type

    `char` Allocates at a minimum 1 byte of memory (per C++ standard)

    - Contains ASCII equivalent when converted to an integer

  - Chars are defined with single quotes:

    ```
    char myChar = ‘A’;
    int charAsASCII = myChar; // Valid
    ```

    - Printing charAsASCII to the terminal will result in the following:

      ```
      cout << charAsASCII << endl;
      ```

      - `65` (Is printed to the terminal)

Lab 2: Introduction to the C++ Programming Language

- C++ Strings
  - Not a ‘built-in’ primitive
  - Built-in C++ standard class
    - Abstraction of dealing with sequences of characters
      - Text, Sentences, etc.
  - Defined with double quotes:

```
string myText = “Hello World”;
```

- Easily printed to the terminal using the `cout <<` operator:

```
cout << myText << endl;
```

- Will display: `Hello World` in the terminal

Lab 2: Introduction to the C++ Programming Language

```
// This program prints to the screen the words:
// PI = 3.14
// Radius = 4
// Circumference = 25.12
#include <iostream>
using namespace std;
const double PI = 3.14;
int main()
{
        float radius;
        radius = 4.0;
        cout << "PI = " << PI << endl;
        cout << "Radius = " << radius << endl;
        cout << "Circumference = " << 2 * PI * radius << endl;
        return 0;
}
```

Lab 2: Introduction to the C++ Programming Language

- 2.1 Working with the `cout` Statement
  - (name.cpp)
- 2.2 Working with Constants, Variables, and Arithmetic Operators
  - (circlearea.cpp)
  - Answer questions asked in exercise 3
- 2.3 Rectangle Area and Perimeter
  - Don’t forget to create the source file:
  - (rectangle.cpp)
- 2.4 Working with Characters and Strings
  - (stringchar.cpp)
  - Answer questions asked in exercise 3 & 4

Lab 2: Introduction to the C++ Programming Language

- Submission File Checklist
  - Submit all files on Canvas. Be sure to include all source files and documents.
  - 2.1 name.cpp
  - 2.2 circlearea.cpp
  - 2.3 rectangle.cpp
  - 2.4 stringchar.cpp
- Don’t forget to answer any questions from the exercises in a comment block at the end of your code